

## Aufgabe 3: josh (12.0 Punkte)

Programmieren Sie basierend auf der vorgegebenen `clash` eine Shell, die in ihrer Funktionalität um Signalbehandlung und Umleitung von Ein- und Ausgabekanal erweitert ist: `josh` (**job shell**). Die unten aufgeworfenen Fragen sind in der Datei `josh.txt` zu erläutern.

### a) Sofortiges Aufsammeln von Zombieprozessen

Die `clash` sammelt angefallene Zombieprozesse jeweils vor der Ausgabe eines Promptsymbols auf. Ändern Sie dieses Verhalten so, dass Zombieprozesse immer direkt nach ihrem Entstehen aufgesammelt werden (**sigaction(2)**). Die Ausgabe des Exit-Status und der Kommandozeile soll unmittelbar nach dem Aufsammeln des Zombieprozesses auf dem Standardfehlerkanal erfolgen.

### b) Signalhandler

Die `josh` soll nun das SIGINT-Signal (Ctrl-C, geht an alle Prozesse des Terminals) behandeln. In der Signalbehandlungsfunktion soll nur die Meldung „Interrupt!\n“ auf den Standardfehlerkanal ausgegeben werden. Was passiert mit dem laufenden Vordergrundprozess und den eventuell laufenden Hintergrundprozessen, wenn Sie Ctrl-C tippen (→ Dokumentation)? Ändern Sie die `josh` nun so, dass die Hintergrundprozesse das SIGINT-Signal ignorieren. Was ändert sich dadurch am Verhalten bei einem Ctrl-C (→ Dokumentation)?

### c) Nebenläufigkeitsprobleme

Durch die nebenläufige Arbeit auf der Jobliste durch Signalbehandlungen und den eigentlichen Programmablauf kann es zu sogenannten Race-Conditions kommen. Identifizieren Sie die möglichen Probleme und Lösungen, beschreiben Sie diese in der Dokumentation und setzen Sie sie in Ihrer Implementierung um (**sigprocmask(2)**).

### d) Standardein-/ausgabekanal umleiten

Zuletzt soll die Shell noch das Umleiten der Standardein- und ausgabe erlauben (**dup2(2)**). In die Standardeingabe wird der Inhalt der Datei `inFile` umgeleitet, falls das Token `<inFile` auftritt. Die Standardausgabe wird in eine Datei `outFile` umgeleitet, falls das Token `>outFile` auftritt. Dabei wird `outFile` angelegt, falls es noch nicht existiert, und überschrieben, wenn es bereits existiert. Das Umleitzeichen `<` beziehungsweise `>` ist hierbei immer jeweils mit dem Dateinamen verbunden. Die Token sollen – soweit vorhanden – immer am Ende der Kommandozeile in der Reihenfolge `<`, `>` und `&` auftreten.

### Hinweise zur Aufgabe:

- Erforderliche Dateien: `josh.c`, `josh.txt`
- Ihre Shell soll mit Optimierungsstufe `-O3` des Compilers funktionieren.
- Die für die Umleitung der Standardausgabe angelegte Datei soll mit Lese- und Schreibrechten für den Besitzer sowie mit Leserechten für die Besitzergruppe und alle anderen Benutzer erzeugt werden (`rw-r--r--`).
- Für die Hilfsfunktionen, die in der Bibliothek `libshell.a` enthalten sind, finden Sie eine Doxygen-Dokumentation auf der Übungswebseite.

### Hinweise zur Abgabe:

Bearbeitung: Einzel  
Bearbeitungszeit: 6 Werktage  
Abgabezeit: 17:30 Uhr